

WUR_SMP_TemplateAndGuidance_v01

Authors: Danny de Koning – van Nieuwamerongen
<https://orcid.org/0000-0002-8264-4541>

Irene Verhagen
<https://orcid.org/0000-0001-5588-1333>

Joost Albers
<https://orcid.org/0009-0003-7170-7871>

Date: 20240112

Organization: Wageningen University & Research
<https://ror.org/04qw24q55>

Licence:



<https://creativecommons.org/licenses/by/4.0/>



WAGENINGEN
UNIVERSITY & RESEARCH



WUR Software Management Plan template

- ❖ Definition of research software (RS): “Research Software includes source code files, algorithms, scripts, computational workflows and executables that were created during the research process or for a research purpose. Software components (e.g., operating systems, libraries, dependencies, packages, scripts, etc.) that are used for research but were not created during or with a clear research intent should be considered software in research and not Research Software.” (source: Gruenpeter et al. 2021, DOI: [10.5281/zenodo.7589725](https://doi.org/10.5281/zenodo.7589725)).
- ❖ Use this template to fill in a Software Management Plan (SMP) for your research project and only for RS that can potentially be reused within other research projects or decision-making processes (i.e. RS of management level medium and / or high; source: Martinez-Ortiz et al. 2023, DOI: [10.5281/zenodo.7038280](https://doi.org/10.5281/zenodo.7038280), page 22-28). It is less relevant for code that is not intended for reuse outside of a specific research project.
- ❖ This template is based on [the WUR Guidelines on Value Creation with Software & Data](#), [Practical guide to Software Management Plans](#), [WUR Information security policy](#), [WUR network regulation](#), and the [WUR privacy policy](#).
- ❖ You are free to add sections to this template to better align with your project. However, the original sections (A-H) must be retained.
- ❖ To get to additional information in the appendix for each section, right-click [\[info\]](#) and select ‘open hyperlink’ or hold keyboard key CTRL + left-click [\[info\]](#).
- ❖ You can also find this SMP template at DOI: [10.5281/zenodo.10473646](https://doi.org/10.5281/zenodo.10473646) and in DMPonline.
- ❖ Any data used to create or test software should be described in a data management plan (DMP). Find the WUR DMP template via DOI: [10.5281/zenodo.7233369](https://doi.org/10.5281/zenodo.7233369) and in DMPonline.
- ❖ For a review of your SMP, contact data@wur.nl.

- ❖ Questions? Contact data@wur.nl or visit the WUR research data management website at <https://www.wur.eu/rdm> for more information. For privacy questions, contact your Privacy Officer (PO) or visit the [Privacy & personal data WUR intranet page](#) (includes a list of POs). For information security questions, contact [your Information Security Officer](#) (ISO) or see the [Information security WUR intranet page](#).
- ❖ Please note that questions can have multiple answers checked where applicable.

A. Describe the research project [\[info\]](#)

1. Name researcher:

2. What is the name of your department or institute?

- Agrotechnology and Food Sciences
- Animal Sciences
- Environmental Sciences
- Plant Sciences
- Social Sciences
- Wageningen Food Safety Research
- Wageningen Food & Biobased Research
- Wageningen Economic Research
- Wageningen Plant Research
- Wageningen Environmental Research
- Wageningen Marine Research
- Wageningen Bioveterinary Research
- Wageningen Livestock Research

3. What is the name of your chair group or business unit?

Please copy-paste the English name and abbreviation for:

- chair groups from [this page](#).
- business units from [this page](#).

Example: Bioprocess Engineering (BPE) or Green Economy and Landuse (GEL).

| |
|--|
| |
|--|

4. Describe the organisational context of your research project.

| | |
|--|--|
| SMP version (or date last modified) | |
| Supervisor / (co-)promotors / project leader | |
| Graduate School (WU only) | |
| Start date of project | |
| End date of project | |
| Project number | |
| Funding body | |

5. Give a short description of the research project including which outputs are expected in terms of research software (RS), and stating its purpose and audience.

| | |
|---------|--|
| Title | |
| Summary | |

6. List the individuals responsible for the following tasks.

| | |
|--|--|
| Software / code development | |
| Git repo owner (if applicable) | |
| Storage and backup | |
| Code archiving / publishing | |
| Research and / or software engineering support | |
| Any other role [.....] | |
| | |

7. I have requested a review of this software management plan from:

- WUR Library – Data Management Support (data@wur.nl).
- The data or model steward of my chair group or business unit.
- A research or software engineer.
- Other, please describe below.

8. Depending on the answer given at question 7, provide the name of the support staff and the date of consultation.

B. Describe the research software (RS) to be created, software used, file formats and size [\[info\]](#)

9. Will you reuse existing RS for this project?

Yes. Please specify in the table below which RS.

| Name of software | Version | Modified or unmodified? | Licence type | Repository link |
|------------------|---------|-------------------------|--------------|-----------------|
| | | | | |
| | | | | |
| | | | | |
| | | | | |

No. Please describe in the text box below any constraints to reusing existing RS.

10. Will new RS be produced?

Yes.

No.

11. Please describe the RS you expect to generate and / or use in the table below. Include reused existing RS as well (as these are materials that you manage and store).

| Materials | Type | Scripting language | (Open) file format | Estimated size of each file (range) | Estimated number of files (range) |
|--|---|----------------------------------|--|-------------------------------------|-----------------------------------|
| (e.g. R cleaning scripts, databases, models, algorithms, packages, etc.) | (e.g. textual, database, data table, etc) | (e.g. R, Python, Java, C++, etc) | (e.g., .R, .py, .csv, .txt, .cpp, etc) | (e.g. 1 – 10 Mb) | (e.g. 50 – 100) |
| | | | | | |
| | | | | | |
| | | | | | |

12. Estimate how much storage you require in total (e.g. by using the information in the table in the previous question).

- 0-10 MB.
- 10-100 MB.
- 100-1000 MB.
- >1000 MB.

13. Please indicate whether you will need computing resources and, if yes, estimate how much you require (in terms of calculating minutes or calculating power).

C. Storage of research software (RS) and documentation / metadata during research [\[info\]](#)

14. Where will the RS and accompanying documentation and metadata be stored and backed up during the research project (see the [WUR Data Storage Finder](#))? Include platforms you use to share or send code for processing, computing, analysis, or collaboration.

- Git@WUR (GitLab locally hosted at WUR).
- WUR high performance computing cluster Anunna (not meant for permanent storage, only computing power and transient storage).
- Linux server or Windows server at WUR.
- M:drive – only when an up to date version of the RS is also safely stored on the W:drive, Git@WUR, or Yoda.
- W:drive (WUR network drive; please specify below whether it is W:drive Enterprise Storage or W:drive Massive File Storage with or without disaster recovery).
- Yoda@WUR (data management platform; SURF hosted WUR instance).
- WUR OneDrive for Business - only when an up to date version of the RS is also safely stored on the W:drive, Git@WUR, or Yoda.
- WUR SharePoint / Teams - only when an up to date version of the RS is also safely stored on the W:drive, Git@WUR, or Yoda.
- Other, please specify below the storage medium / system and describe back-up frequency, access management, and geographic location (e.g. within or outside the EU).

D. Structuring research software (RS) and information [\[info\]](#)

15. Give a (visual) representation of the folder and / or Git repository structure you intend to use.

16. Describe the file naming conventions you intend to use. Please give one or multiple example(s).

17. How will you distinguish between versions of files?

- We will use Git versioning (strongly advised).
- Dates within file names are updated when files are modified (with the exception of Git files when applicable).
- A version number in the format 'v01' will be added to all file names (with the exception of Git files when applicable) which increases after file modification.
- Other, please specify below.

E. Research software (RS) documentation and quality [\[info\]](#)

18. Describe below what [documentation](#) and metadata will accompany the RS and how they will be provided in the textbox.

- The WUR readme file template (see template at <https://doi.org/10.5281/zenodo.7701727>) or equivalent in Git.
- The Yoda metadata form (see Yoda metadata editor at <https://utrechtuniversity.github.io/yoda-portal/>) or equivalent in Git.
- The CodeMeta metadata form (see CodeMeta metadata editor at <https://codemeta.github.io/codemeta-generator/>) or equivalent in Git.
- The WUR codebook template (see template at <https://doi.org/10.5281/zenodo.7701727>).
- Other ([see examples in guidance](#)), please specify below.

19. Describe what quality controls will be used?

- Software developers or research (software) engineers will provide expert knowledge on developed code.
- Statisticians will be consulted.
- Model auditors will audit models. Please specify below the extend of auditing (e.g. A, AA, code review, internal or external review, etc.).
- Supervisors or peers will review the RS and results for any anomalies (e.g. unexpected inconsistencies, incorrect functioning of code, correct labeling, correct and consistent coding applied, etc.).
- We will use standardised labelling and terms of RS so that consistency in coding files and scripts are attained.

We will use discipline specific community standards for labelling and coding of RS. Please specify below the community standard used.

We adhere to RS quality guidelines. Please specify below which ones.

We will perform a [model self-assessment](#).

Other ([see examples in guidance](#)), please specify below.

F. Security, working with sensitive materials (personal data, ethics), ownership / rightsholder(s), sharing and access [\[info\]](#)

20. Who is the owner / rightsholder of the RS?

- WUR is the owner / rightsholder of the RS.
- WUR is not the (only) owner / rightsholder of the RS and a WUR approved formal (consortium) agreement or contract between WUR and other parties is present.
- Other, please specify below.

21. Is this project registered in SmartPIA?

- Yes.
- No. Please register in SmartPIA in the case of (privacy) sensitive issues (when applicable: via your supervisor, the project manager, [see guidance](#)).

22. What is the [classification](#) for your project (for example as specified in SmartPIA) taking into account the (privacy) sensitivity of code or the resulting output of the RS?

- Negligible.
- Some.
- Serious.
- Disruptive.

23. Please specify the protection measures in the case of (privacy) sensitive issues.

Only WUR provided storage, processing, and analysis platforms are used as consulted with the ISO and PO where applicable.

Access management to the RS is either managed or approved by the project leader / supervisor of the project and contains clear documentation of who has access.

Informed consents are present when privacy sensitive information are involved.

Personal or other sensitive information will be removed from RS (code) when not required for verification of research.

Personal and other sensitive information will not be made openly available and will at most be shared under formal agreements for which the legal department, ISO and PO are consulted.

We will consult with the ISO and PO for appropriate measures to undertake (highly recommended).

Other ([see examples in guidance](#)), please specify below.

24. Describe the ethical risks that should be considered by developers and users of the RS and / or its output, such as privacy, information security, ethical issues, which may include approval from [ethical committees](#)?

No risks associated with RS development or usage of RS.

I work with animals and will seek / have approval of the ethics committees involved (Animal Welfare Body (IvD), Animal Tests Committee (DEC), Central Animal Testing Committee (CCD)).

I work with humans in a social sciences aspect or that doesn't fall under Medical Research Involving Human Subjects Act (WMO) and will seek / have approval of the ethics committees involved (WUR Research Ethics Committee).

I work with humans in a biological / physiological / medical / intervention aspect falling under the Medical Research Involving Human Subjects Act (WMO) and will seek / have approval of the ethics committees involved (Medical Ethics Committee-East (METC-Oost)).

I work with genetic resources covered by the Convention on Biological Diversity and need to comply with the Nagoya Protocol.

The RS output could negatively impact nature, wildlife conservation efforts, humans, society, policy creation or decision making. Please explain below.

The RS will become publicly available and as such could be a target for misuse (for example misuse of computing power for mining cryptocurrency or as part of a spam farm, underlying sensitive data may be breached, etc.). Please explain below how you mitigate the risks.

Other, please explain below.

25. Will there be any intellectual property (IP) rights or alternative applications or routes to impact (such as commercial interests) associated with the RS?

No.

Yes. Please explain how IP rights will be managed and who will be contacted to discuss such rights handling.

Maybe / yes. We will need further support or we will fill out a Data & Software Disclosure Form (DSDF) to find out the possibilities for value creation and protection.

G. Research software (RS) archiving and publishing [\[info\]](#)

26. Are there reasons to restrict access to the RS or limit which RS will be made publicly available?

- No.
- Privacy / GDPR.
- Ethics.
- Contractual agreement.
- Commercial interests.
- Public security.
- IP rights.
- Other, please specify below.

27. Describe what materials from section A will be archived at WUR (e.g. WUR network drive / Yoda) and not published? Include the exact name of the storage medium chosen (see [WUR Data Storage Finder](#)).

- Not applicable as RS will be made openly available.
- Due to sensitivity, we will need to archive (part of the) materials underlying articles or reports internally. Please specify below which materials and the chosen storage medium.
- Other, please specify below.

28. What materials from section A will be published and made available for reuse via a repository or Git platform?

RS underlying publications or reports. Please specify below which materials listed in the table in section A.

Only the metadata is published in a repository as the RS are too sensitive to openly share.

RS not underlying an article or report will also be published. Please specify below which materials listed in the table in section A.

Other, please specify below.

29. When will the RS be available for reuse, and for how long will it be available?

RS will be available as soon as the article or report is published and not required for any other article publication. Please specify below the duration of availability.

RS will be available upon completion of the project. Please specify below the duration of availability.

Published RS becomes accessible after expiring of the embargo. Please specify below the reason for an embargo and the duration of availability.

Publication of RS not underlying an article or report will be considered at the end of the project.

Other, please specify below.

30. How will the RS be packaged? Please provide a link to available packaging information (e.g. entry in a packaging registry, if available).

No packaging required, RS can simply be installed or downloaded from Git.

MSI.

App-V.

Python package distribution.

R packaging.

Other ([see examples in guidance](#)), please specify below.

31. How will reading, writing, and exchange of data by the RS be ensured following domain-relevant community standards?

The RS is able to read in and export to preferred formats such as displayed on the [DANS Data Stations preferred format list](#).

The RS only reads and exports software or machine specific files that are proprietary and is unable to use or produce preferred formats. Please specify below why.

The RS reads in community standard files that are commonly used and available within our specific research discipline. Functions to read in preferred / non-proprietary formats will not be provided.

Other, please specify below.

32. How will you ensure that the RS is operating system (OS) independent?

We will not make OS independent RS. The RS is as is and will only be able to be used as packaged for the OS in question or code will only be tested on the OS used in case no packaging is applied.

We will create different packages for different OS and test them accordingly or we will test our code on different OS in case no packaging is applied.

Other, please specify below.

33. Where will the RS be made (publicly) available (see also the [WUR Repository Finder](#))?

Git@WUR.

4TU.ResearchData, DANS Data Stations, or Zenodo will be used to publish a Git repository and issue a DOI.

PyPi repository (Python).

CRAN (R).

Other ([see examples in guidance](#)), please specify below.

34. Which metadata standard will be used to describe the RS when archiving at WUR and / or depositing in a repository?

- Yoda metadata through the platform itself or the public [Yoda metadata editor](#).
- The [CodeMeta metadata](#).
- Metadata standard from the DANS Data Stations, 4TU.ResearchData and / or Zenodo.
- We will register in Pure via data@wur.nl (mandatory for WUR research output).
- Other ([see examples in guidance](#)), please specify below.

35. Which [licence](#) will be applied to the RS?

- MIT Licence.
- Apache Licence 2.0.
- GNU GPL Licence (specify below which version).
- Other ([see examples in guidance](#)), please specify and / or provide link to the licence below.

H. Research software (RS) management costs [\[info\]](#)

36. What resources are required for developing the RS during the project?

- Our entire team will contribute to RS development for which the costs are covered by the project or group.
- We will require support from dedicated research engineers / software developers for which the costs are covered by the project or group.
- The RS is made open and development will be a community effort. Ownership / rightsholdership will be discussed and determined before making code publicly available.
- The researchers in the project will spend at least 10% of their time on RS management to approach the FAIR principles as much as possible.
- A model auditor will audit the research model for which the costs are covered by the project or group.
- Other, please specify below.

37. What resources are required for maintaining the RS after the project?

- The RS is not eligible for further maintenance as it is either not openly available or we simply do not provide any further maintenance (the RS is 'as is').
- Project members will contribute to maintaining the RS within Git for at least the duration of their employment even after the project has finished.
- The RS is made open on a Git platform and maintenance will be a community effort. A WUR employee will be assigned Git project repository admin rights to maintain the repository and approve merge requests.
- When maintenance or support to the RS is not provided anymore, this will be clearly documented within the Git project repository and any other publications of the RS.

Financial costs of WUR employees in providing time for maintenance and development is carried by the group after publication and after the end of the project.

Other, please specify below.

38. What resources are required for running the RS during and after the project?

The RS only relies on storage in Git and is covered by our research group or WUR.

The RS relies on CI/CD, running jobs, or other actions requiring computing power / servers, for which the costs are covered by the research group during the project and for several years after publication.

The RS relies on authentication processes integrated in the WUR single sign on system / identity provider.

Output of RS is pushed to specific websites. Please specify below the web address and the owner of the domain.

Other, please specify below.

39. What resources are required for providing user support for the RS during and after the project, and how will it be organised?

We will not provide user support.

Issues can be provided within our Git instance which will then be attempted to be resolved by the project team or community.

A dedicated email address will be made available for support requests which our group will answer.

We require dedicated institutional support services. Please specify below what is required, who will provide the support services, and how costs are covered for these services.

Other, please specify below.

Guidance: additional information (may be deleted after completion)

A guidance. Describe the research project.

1. Name researcher

Please add your full name.

2. Name department.

Please choose your department. If you are working from multiple departments, you can choose multiple answers.

3. Name chair group or business unit.

Please add the English name of your chair group or business unit exactly as specified on the [chair group](#) or [business unit](#) webpage. Please carefully check the spelling. Preferably copy and paste directly from aforementioned websites.

4. Organisational context of your research project.

An SMP should always include contextual information about the researcher and the project. Then, it is clear to whom and which project the filled in SMP belongs and to which RS the described management practices apply.

5. Description of your research project.

Giving a short description of your research helps the reader to understand your work and put the description of your RS management into context.

6. RS management responsibilities.

Identifying persons who play a role in your daily RS management practices helps clarify the RS development process. Identifying these roles are also important in the event you leave WUR, or on completion of your PhD. For RS to be accessible for several years, responsibility for the preserved RS should lie with more than one person. Feel free to add more roles.

7. Requested review.

You can request a review of the SMP from data@wur.nl and / or from a research or software engineer, a model steward, or your local data steward. Provide the contact details of the reviewer in the next question.

8. Support staff.

Provide the name of the support staff contacted for review of this SMP (the one mentioned in the previous question).

B guidance. Describe the research software (RS) to be created, software used, file formats and size

9. Reuse existing RS.

When others make RS available (e.g. through Git or publication) it gives you the opportunity to reuse RS within your research. However, make sure that you are aware of the terms of use (e.g. the licence) of the RS that you want to reuse. The terms of use provide you with what you are allowed to do with the RS.

In the case of not reusing RS, think of potential reasons why the reuse of existing RS was considered but not implemented. Is the RS you want to reuse not publicly available or otherwise restricted in access? Would the costs be too high (e.g. acquiring specific software, cloud computing services required, or too high of computing units required)? Or is there (currently) no RS available that fits your purpose?

10. New RS.

Indicate whether you will develop new RS in the research project.

11. Description developed RS.

Type refers to whether the RS material can be classified as for example textual, database, data table, etc. Try to use software / scripting languages that let you also save files in an [open format](#), so that others can open the files (now and in the future) even if they don't have the software. Providing the estimated number of files and file size helps to indicate for example the storage requirements and costs.

12. Storage volume.

According to the table in the previous question, estimate the amount of storage space you expect to need.

13. Computing resources.

Please indicate whether the RS requires any platform computing resources and how much is required (e.g. Git CI / CD minutes, running execution jobs on Git platforms, high performance computer cluster resources, virtual environments / machines, etc.).

C guidance. Storage of research software (RS) and documentation / metadata during research

14. Storage RS and documentation.

Ensure that you store RS and accompanying documentation and metadata safely in compliance with the [WUR IT security policy](#) and the [WUR Guidelines on Value Creation with Software and Data](#). [Safe storage solutions](#) would be for example the solutions indicated in the [WUR Data Storage Finder](#), possibly in combination with IT-approved cloud storage. Note that if you store RS and accompanying documentation and metadata with a WUR cloud service or on the WUR network (with the exception of W:drive – Massive File Storage), they are backed up automatically. Contact your Information Security Officer for help in determining appropriateness of other storage or platform solutions when developing RS.

D guidance. Structuring research software (RS) and information

15. Folder structure.

Designing a logical folder structure (for tips, see [here](#)) ensures that you and fellow researchers can easily locate required elements of developed RS now and in the future, and helps in easier maintenance of the RS. Provide how you plan to organise your folders or Git structure. If you already have a folder structure in place, you can easily create a folder tree structure in windows using PowerShell (note: the following steps do not work with zip files):

- Go to your file explorer.
- Open the parent folder (or Main local Git directory).
- In the address bar, type and press enter afterwards: powershell
- A blue screen pops up. Continue in that screen.

- Type and press enter afterwards: `Get-ChildItem | tree > foldertree.txt`
- There is now a file in your folder [foldertree.txt], from which you can copy the content in the answer field.

If you're using Git Bash on Windows, go to your parent directory (or Main local Git directory) and use:

- `cmd //c tree //a > foldertree.txt`
- There is now a file in your folder [foldertree.txt], from which you can copy the content in the answer field.

If you are using Linux:

- `sudo apt install tree.`
- Go to your project parent directory.
- `tree -d > foldertree.txt`
- There is now a file in your folder [foldertree.txt], from which you can copy the content in the answer field.

16. File naming conventions.

Applying a consistent and descriptive file naming convention (i.e. a systematic file naming method) helps to:

- identify the content of a file without opening it.
- easily and quickly locate, retrieve and filter files, even if they have changed folders.
- easily sort and browse through your files.
- identify missing files.

A basic example of a file name:

[project]_[filesubject]_[subsubject]_[date]_[version].[extension]

Using the international standard for date (YYYYMMDD) and applying a leading zero to the version number (e.g. v01) aids in ordering files. Find tips for naming files [here](#).

17. File versioning.

Make sure that you have a system in place to keep track of file versions. A simple and effective system is to incorporate version numbers (e.g. v01, v02 etc.) in your file names. Note: for files in Git, dates and version numbers should not be added to the filenames. Git is version control software, keeping track of versions and dates for you and without having to apply file versioning yourself. When applying versioning by changing file names and dates, Git will not recognise a new version as such, but as a completely different file.

E guidance. Research software (RS) documentation and quality

It is essential to systematically document RS during your research, whether done by archiving at WUR, making RS available through Git, or depositing RS into a repository. Documentation is information added to RS to ensure that it is understandable and reusable to yourself and to others, both during and after your research. Metadata is also documentation, but in a structured form, describing the RS in a way that facilitates cataloguing and discovery.

18. RS documentation and metadata.

The most common required form of documenting RS is by adding a readme file, which is further supplemented with metadata. Independently of where RS is preserved (e.g. at WUR, in a repository) it should at least be accompanied by:

- A readme file

The readme file contains information about e.g. any steps undertaken in developing the RS or which are required to use software, installation and or usage instructions, any dependency requirements, etc. In short: all information necessary to understand, implement, and reuse RS. WUR Library advises to use the WUR readme file template (DOI: [10.5281/zenodo.7701727](https://doi.org/10.5281/zenodo.7701727)) as the minimum required documentation to add to the RS. Feel free to add more documentation where appropriate and required. When Git platforms are used, please use the .md extension instead of a .txt extension (or have the Git platform generate an empty readme.md file for your git project, which you can then fill in).

- Metadata

Metadata is machine-readable information about the RS, according to fixed terms, which makes the RS findable and searchable. If you are not publishing your code in

a repository or are not making the RS available through Git platforms, then WUR Library - RDM Support advises to use the Yoda metadata terms as the minimum required metadata to add to the RS. You can fill in these terms in Yoda, when applicable as a storage solution, or use the public [Yoda metadata editor](#) and download the metadata as a .json file. When you make the RS available through publishing in a repository, then follow the metadata fields of that repository. If you are only making RS available through Git (unadvised to do so as the only option), make sure you add 'topics' to your Git project repository (in the settings area of your Git project repository) to help findability of the RS. Additionally, WUR Library - RDM Support advises to use the [CodeMeta metadata](#) as well to be added.

- A codebook

WUR Library advises to use the WUR codebook template (DOI: [10.5281/zenodo.7701727](https://doi.org/10.5281/zenodo.7701727)) to explain variables, abbreviations, etc, in .csv format. This enables users to get a structured overview of abbreviations and meanings of variables used.

- Other examples

Other forms of documentation can be, but are not limited to:

- Elaborate documentation and notes within scripts / code.
- Installation documentation.
- Dependency documentation (e.g. dependency on other software, packages, data).
- Software contribution guidelines.
- Citation information in the form of a software citation file format (CFF).

19. RS quality controls.

Describe the performed practices to ensure the quality of the RS. RS can be checked for completeness, sufficient documentation, correctness, unnecessary duplication, (statistical) anomalies, efficiency, etc. This can include practises like code validation, peer review of code, the use of controlled vocabularies, and [self-assessment of models](#).

More specifically, quality controls can entail, but are not limited to:

- Standard code reviews via merge requests in the chosen Git platform.
- A community driven effort is made for quality control using Git platforms.
- Using CI / CD pipelines to deploy code.

- Employ various environments for development, testing, acceptance, and production (DTAP environments).
- Thorough testing of the RS (when applicable, specify how).

F guidance. Security, working with sensitive materials (personal data, ethics), ownership / rightsholder(s), sharing and access

When RS includes sensitive aspects, a considered approach is required for code sharing, storage, and use, to mitigate any security, public image, operational, financial or privacy risks (see [data classifications](#)). For privacy questions, please consult your Privacy Officer ([WUR intranet page Privacy & personal data](#) – login required). For information security questions, contact [your Information Security Officer](#) (login required) or see the [WUR intranet page information security](#) (login required).

20. Owner(s) / Rightsholder(s) and access.

The party that is the rightsholder / has ownership over the RS decides what others are allowed to do with it. WUR's policy on IP & Value Creation (see [here](#), intranet WUR, login required, under 'Related files') stipulates that WUR is rightsholder of RS developed by its staff. However, if you develop RS from / with an external party, you may have other contracts or agreements (e.g. consortium agreement, (data) sharing agreement) which overrules WUR's ownership policy. Explain which parties are (joint) rightsholders / have (joint) ownership and thus control code sharing and access.

21. Registration in SmartPIA.

When (privacy) sensitive information is processed when developing RS (including when such information is used in lines of code), the project needs to be registered in WUR's central reporting register for (privacy) sensitive data (SmartPIA). Check (with your supervisor, project manager, or related DMP) if this project has been registered in SmartPIA. If not registered already, the project manager needs to register your project via the blue button on [this WUR intranet page](#) (login required). Based on the type of personal data used and the purpose of that use, the project manager will be notified if any safety measures need to be taken.

Note: PhD candidates, when applicable, do not have to register their projects themselves. Please, contact your supervisor and / or project manager.

22. Classification.

Classifying the RS according to the [WUR data classification scheme](#) aids in determining the potential impact or damage the RS may cause to WUR and individuals when there is loss of availability, integrity or confidentiality. This impact or damage depends on the sensitivity of the RS, e.g. by using personal data, (financial) company information, governmental information, algorithms that discriminate or identify individuals, societal sensitive algorithms, etc. The classification consequently aids in determining which (security) measures need to be applied and which steps to take when handling, sharing, and storing the RS. Your Information Security Officer (ISO) and Privacy Officer (PO) can help you with the data classification.

23. (Privacy) sensitive protection measures.

Working in accordance with the WUR policy, laws, privacy law, or GDPR, requires careful processing and storage of (sensitive) personal information. Risks should be mitigated by using data minimisation, informed consent forms, managed access to the RS, secure storage (IT managed) and transfer (encryption), pseudonymisation, and anonymisation. See [WUR intranet page Privacy & Personal Data](#) (login required) for more information about personal data and who is the Privacy Officer (PO) of your Science Group.

Other protection measures specifically applied to RS can be, but are not limited to:

- RS users require authentication (e.g. multi factor authentication).
- Passwords or keys are not stored in version control platforms and are not hardcoded in scripts.
- Automated deployment pipelines are used to limit access to production systems. Access to production systems is always limited.
- Vulnerability scanning is done on the code and on dependencies.
- Penetration tests are performed on production systems.
- Data is encrypted in the data storage solution (such as a database and on disk).

24. Ethical risks.

Here, consider whether you need ethical clearance from e.g. an animal experimental committee or a medical or social ethics review committee. Take into account whether your RS could be used for nefarious purposes or have negative consequences on animals, nature, or society.

25. Intellectual Property (IP) rights associated with the RS.

Indicate whether IP rights are affected. If so, explain which and how they will be managed. Think about the licence (how others can use the RS) and a potential embargo period, which means that others can see that the RS, but is made available later (e.g. when you want to publish findings or a patent first). Note: when RS or data with potential commercial or utilisation value is created, it should be disclosed by completing a [Data & Software Disclosure Form](#) (DSDF) (source: [WUR guidelines on Value Creation with Software and Data](#), support at valuecreation@wur.nl).

G guidance. Research software (RS) archiving and publishing

WUR requires that all research materials underlying publications required to verify results (including RS) is preserved. RS should be made publicly available for reuse, unless there are valid reasons not to do so. In that case, RS (including documentation and metadata) should at least be archived within WUR.

Is this project funded by NWO? If yes, NWO expects researchers to preserve research output and materials resulting from their projects for at least ten years, unless legal provisions or discipline-specific guidelines dictate otherwise. As much as possible, research output should be made publicly available for reuse. As a minimum, NWO requires that the research materials underpinning research papers should be made available to other researchers at the time of the article's publication, unless there are valid reasons not to do so. The guiding principle here is 'as open as possible, as closed as necessary'. Due consideration is given to aspects such as privacy, public security, ethical limitations, intellectual property rights and commercial interests. NWO recognises that software may be necessary to access and interpret data. In such cases, the software management plan will be expected to address how information about such items will be made available.

26. Reasons to restrict access or limit public availability.

It could be that (a part of) the RS cannot be made publicly available or that access to the RS should be restricted. For example, because (part of the) RS contains personal data, there are commercial interests involved, or making the RS publicly available could infringe on public security. When in doubt contact data@wur.nl or your Privacy Officer (PO) and / or Information Security Officer (ISO).

27. What RS materials are internally archived?

It is not always possible to publish RS in a repository due to legitimate reasons. However, that does not mean that the RS should not be preserved. Additionally, unpublished RS that has not been used in a publication can still have (future) value for reuse.

28. What RS materials will be published?

It is advised to not only make RS available through Git platforms (DOIs not offered), but also publish the RS in a repository to enhance the findability. A repository is an infrastructure accessible via an online platform, which manages, preserves, shares and provides access to data, RS, metadata and documentation. They furthermore assign a persistent identifier (such as a DOI). If you release various official versions in your Git platform, publish those versions in your chosen repository as new versions as well (check whether your repository allows for DOI versioning). If you cannot publish the RS in a repository, you can publish the metadata. Because metadata is machine-readable, the existence of the RS can be discovered, adhering to the 'F' (findability) of the FAIR principles.

Is this project funded by ZonMw? If yes, ZonMw expects researchers to publish the metadata in the case the research materials themselves cannot be published due to e.g. privacy, public security, ethical limitations, intellectual property rights and commercial interests.

29. When and for how long will the RS be available for reuse?

The data policies of WUR and NWO require, as a minimum, that research materials underlying publications that are required for verification of results, should be made available to other researchers at the time of publication (unless there are valid reasons not to do so) and for several years. Indicate when the RS will be made available. Explain the reason of any embargo periods. Explain whether exclusive use of the RS will be claimed and if so, why and for how long. Indicate whether RS sharing will be postponed or restricted for example to publish, protect intellectual property, or seek patents.

30. RS packaging.

In order for others to reuse the RS, it may need to be made available in a packaged state (compiled and installable). Please indicate whether packaging of the RS will be performed and in which form. Research or software engineers may be available to

help in packaging software. Additional examples of packaging types are APPX, APK (Android), Debian, RPM, Flatpack, Tar archive (tar.gz).

31. Reading, writing and exchanging of data in community standards.

When RS is able to handle different types of preferred formats, the interoperability and reusability increases. Consider making sure that the input and output of RS follows community standards that are widely used within the scientific discipline or follow the [DANS Data Stations list of preferred formats](#).

32. Operating system independence.

If RS can be used regardless of operating system used (OS), the interoperability and reusability increases. Consider making the RS independent of OS and test the RS on various OS.

33. (Public) availability of the RS.

Which repository to choose depends on for example the required level of access to the RS, whether RS requires installation (and installation of dependencies), and / or if a discipline-specific repository is available. WUR Library [supports](#) the multi-disciplinary repositories [DANS Data Stations](#), [4TU.ResearchData](#) and [Zenodo](#), which means that data librarians can help publish RS and documentation in these repositories. Additionally, deposition costs For DANS Data Stations and 4TU.ResearchData are covered by WUR Library up to a 100GB. However, another repository could be chosen as well (e.g. a public version of GitHub or Gitlab). The [WUR Repository Finder](#) can help you make a decision about which repository to choose to deposit RS. Choose a repository that is certified, that assigns persistent identifiers (e.g. DOIs) to RS and the metadata, and that offers the [licence](#) you want to put on your RS (this may differ between repositories). Additionally, it is recommended to use a repository that is (often) used in your discipline, as such a repository is familiar with the file type(s) and volume within your discipline. Finally, it is advised to use Git platforms next to publication of RS to allow for continued maintenance, development, and community contribution and issue handling.

34. Metadata.

Metadata is highly structured, machine-readable information to describe research output such as RS. A metadata standard is a set of fixed fields to facilitate findability and searchability of research output. Common fields to describe RS are author, year, title, spatial coverage, etc. Many repositories use an existing metadata standard,

such as [Dublin Core](#) or [DataCite](#). This way, any deposit to the repository will use that as a standard. However, when not publishing RS (due to legitimate reasons), you should always add relevant metadata (and publish it). WUR Library advises to use the Yoda metadata and CodeMeta terms as the minimum required metadata to add to the RS when not publishing. For Yoda, you can fill in these terms in Yoda, when applicable as storage solution, or use the [Yoda metadata editor](#) and download the metadata as a .json file. For CodeMeta you can use the [CodeMeta generator](#) and download the metadata as a .json file. Pure registration is mandatory for WUR research outputs and this option should always be selected. Please send an email to data@wur.nl for help in registering in Pure. Registering the RS in the [WUR Model Gallery](#) or eScienceCenter [Research Software Directory](#) can be an added benefit to increase findability of the RS. However, if there are discipline-specific metadata standards available, these are recommended.

35. Licences.

A license defines the terms of use for research output to reusers. As such, applying a license to developed RS informs reusers what they are allowed to do (or not) with the software regarding e.g. (commercial) use and relicensing. Indicate under which license the RS may be reused. There is a large variety of software licences available, e.g. (in addition to the provided option) GNU AGPL, GNU LGPL, one of the BSD licences, etc. Creative Commons licences are not suitable for licensing RS, because

- They do not contain specific terms about the distribution of source code.
- They are currently not compatible with major software licences, which makes it difficult to integrate CC-licensed work with other free software.
- Although CC 0 is compatible with many software licences, it is not recommended by the [Open Source Initiative](#).

There are several tools and guides available to aid in making a choice, for example: the [GNU how to choose a license guide](#), the [GNU various licenses guide](#), choosealicense.com, or the [Open Source Initiative](#).

If parts of the RS rely on other software, then one needs to take into account the license on the other software. Please check [WUR Guidelines on Value Creation with Software and Data](#) table 3 (page 21) for compatibility of new licenses with existing licences. If no commercial value or contractual restrictions, use an Open Source Software (OSS) licence.

H guidance. Research software (RS) management costs

36. Resources required for developing RS during the project.

Think about investment in money, time, and infrastructure when it comes to developing and publishing RS. Take into account the work required during research for proper handling, storage and documentation, as well as the work after research to make your RS as FAIR as possible.

37. Resources required for maintaining RS after the project.

Think about investment in money, time, and infrastructure when it comes to maintaining RS (maintaining functionality, handling errors, updating dependencies, updating code, etc). Take into account that multiple people may need to perform such actions for several years.

38. Resources required for running RS.

Think about investment in money, time, and infrastructure when it comes to requirements for running RS (computing units, cloud virtual environments, high performance computing clusters, etc.).

39. Resources required for user support.

Think about investment in money, time, and infrastructure when it comes to supporting users in using the RS. This could be limited to only handle Git issues or have a dedicated email address that allow users to contact you directly.